Machines and Humans Languages

By Gabe Wilberscheid

Since the dawn of computer science, researchers have wonder how machines and humans will communicate. This is famously shown in the Allan Turnings test for Artificial Intelligence (AI), the Turning test. The Turning test consists of a computer and a human having a conversation, except the human does not know it is talking to a computer. Turing hypothesized that if a human cannot tell it is talking to a computer, then we can deem the computer a thinking machine (Turing, 1950). What Turing and many other AI researchers assume is that to be intelligent a machine much speech in languages that we understand. This is a bold assumption. What makes us think that any intelligent species will communicate the same way we do? It would be absurd to think that if an advanced alien species came down to earth today that it would speak English as we do. Yet almost all modern humans judge the intelligence of a machine not based on the complex math it can do, nor the predictions and forecasting it can do, but on how well it understands us when we speak to it. If Siri or Alexa misunderstand you or give an inappropriate answer to a request people use it as an example of the unintelligence these machines have. What I have learned during this project is not so much that machines cannot understand human languages or that machines cannot "think" for themselves. But rather when machines talk and think, it is in ways humans cannot understand.

My research began with an evaluation of the state of the art in Natural Language Processing (NLP), a field that has been around in some form since the advent of computers. A reoccurring problem has been how to encode human languages into computers, as human languages consist generally of alphabets, although many languages have no alphabets and instead use a logographic (a different symbol per word). And of course, there are languages that use neither an alphabet or a logographic and use some combination of the two. The difficulty, of course, comes from the fact that computers only have two states of memory, 0's and 1's. The question of how one maps these finite alphabets or symbols in a logographic to a machine language that only uses 0's and 1's. Early methods included mapping each letter to a number using some encoding like ASCII. While this word in the fact that the machine can then read in the human languages, there is a major flaw. Humans do not learn how to carry out everyday communications by paying attention to the individual letters that come in a stream of sentences. The focus on words and their meanings in the real world. Around this time Noam Chomsky published his groundbreaking research on formal languages. Chomsky had formalized a notation used in syntactic analysis and had categorized languages into a hierarchy of four types of grammars. Each grammar was used to create a language and each language had a type of automation that could parse it (Chomsky 1956). Chomsky would go onto revolutionize the field of linguistics. What was begging to be realized is that humans do not have such a strong grasp on the languages they use themselves.

Next came the realization that rather than encode letters into a machine, we must teach it words. This worked better than using just letters. Once again, however, researchers ran into problems. While they were able to encode a language of n-words into a one-hot vector of n-1 (where each location in the vector represents a word, that place is a one, all other vector columns are a zero), this method was not without its flaws. For one, the vectors are extremely sparse, meaning there is not a lot of information encoded into them. Secondly, for the computers at the time loading, even a small language of a few hundred words was very computationally expensive. This time was marked by many well-funded teams trying to encode handwritten instructions into machines, this became known as rule-based language processing systems. The fatal flaw in this type of system is that there is an infinite number of correct and coherent sentences that can be formed in any language. Chomsky had shown this during his continued research. Thus creating a ruled based system that can parse and stem any sentence in a language is impossible.

Another aspect that made these rule-based systems so limited was the way in which words were encoded. Using the one-hot encoding, the machine had no understanding of the meaning of the words it was using, nor could it tell that there was a relationship between some words and not others. For example, the words cat and dog would be to entirely different one-hot vectors that the machine could not tell have some sort of relationship, they are both animals and both common pets. This problem made it difficult for the machine to generalize meaning in words that humans intuitively know have some sort of relationship.

A paradigm shift came in the late 1980s and mid-1990s when researchers began using machine learning in new ways in regards to NLP. Machine learning had been around for several decades, however, it was not until researchers began using the statistically-based machine learning methods to learn inference and relationships between words that it helped leap the field of NLP forward. This was drastically different than what had been done using rule-based models. Statistical based model did not rely on the programmer to hand program rules into the system that they saw fit, rather it looked at large bodies of text, a corpus, to learn statistically-based rules for meaning and understanding. These systems did well on learning common meanings and did a better job handling misspelled words or uncommon phrases of speech compared to the handwritten rules that had come before it.

As the field of machine learning continued to advance, NLP used new-found methods and models to achieve state of the art results in many tasks. A hallmark of these advances in machine learning came from the use of artificial neural networks (ANN). Which had been invented in 1958 by psychologist Frank Rosenblatt, but had gone through a "neural network winter" where researchers though the limits of ANN's had been reached. It the early 1990s several young researchers began using larger datasets and more powerful computers to create new ANN architectures that would go on to revolutionize the field. One of the researchers was Yoshua Bengio, who used ANN to create high dimensional word vectors. This method used an ANN to look at huge corpora of billions of words and create 300-dimensional vectors for each word.

Each dimension adds some sort of meaning according to the ANN, and while some dimensions were recognizable by humans, many were not (Bengio 2006). The beauty of this method is these word vectors have a ton of information and meaning about the word encoded within them. These word vectors are then fed into a different ANN to train it on more specific tasks. The real elegance from these trained word vectors comes from the ability to do simple operations on them and get astonishing results. For example, if you take the word vector for king and subtract the word vector for man, and then add the word vector for woman, you will get a word vector that almost perfectly matches the word vector for queen. All this is done without explicitly programming any rules into the model.

As a final part of my project, I built a naive text generator. Which is capable of being trained on a piece of text, then given a sample text, can output novel text similar in style to the text it was trained on. I used the Python libraries Keras for deep learning and Spacy for NLP tasks. The model I built was a character-level model, meaning it was fed characters, not words. Still, after just a few hours of training, it was able to spell many words correctly and string a few coherent words together. Here is some output given an input text form my model that was trained on Alice in Wonderland.

Input text: "an angry voice the rabbit s pat pat whe"

Model output:"an angry voice the rabbit s pat pat where what they worst alice stilld poise beloud beight but thing was a causi there s tring alice who said the finish was know decl but the jury was at least when they had to know the middle and when she"

What I took away from my project is that natural languages are highly complex and even humans are just in the early stages of formally defining the rules by which they must follow. The biggest advances in the field of NLP came when humans quite trying to handwrite rules of language to the machines and instead let the machines learn their own rules. First with statistical NLP and then with trained word vectors. While machines have a long way to go before they are writing novels like great authors, they seem to have a much stronger understanding of the languages we use than most people give them. So next time you curse Alexa for not understanding you, realize how far these machines have come in such a short time and ask yourself if understanding human language is the only way to demonstrate intelligence. After all, according to this logic, songbirds see all humans as blabbering fools.

Sources

Bengio, Yoshua; Schwenk, Holger; Senécal, Jean-Sébastien; Morin, Fréderic; Gauvain, Jean-Luc (2006). *A Neural Probabilistic Language Model*. *Studies in Fuzziness and Soft Computing*. **194**. pp. 137–186. doi:10.1007/3-540-33486-6_6. ISBN 978-3-540-30609-2.

Chomsky, Noam. *Aspects of the Theory of Syntax*. M.I.T. Press, 1965.

Smeaton, Alan. (1992). Progress in the Application of Natural Language Processing to Information Retrieval Tasks. The Computer Journal. 35. 10.1093/comjnl/35.3.268.

Turing, A. M. "I.—Computing Machinery And Intelligence." *Mind*, LIX, no. 236, 1950, pp. 433–460., doi:10.1093/mind/lix.236.433.